一种面向低轨卫星网络的高效无证书身份认证方案

张　毅，吴　奇†，周霜霜，贾梦朝
(重庆邮电大学 通信与信息工程学院，重庆 400065)

摘　要：针对现有低轨卫星网络认证方案采用集中认证方式认证存在时延大和采用复杂的双线性映射存在计算开销大的问题。引入无证书认证模型，在 Gayathri 方案的基础上，设计了一种高效无证书认证方案。该方案将用户的公钥和真实身份统一起来，使得认证过程中不需要第三方参与，降低了认证时延；通过椭圆曲线上少量点乘和点加运算构建认证消息，避免使用双线性映射，降低了计算开销。并在随机预言模型下，基于椭圆曲线离散数学对问题假设对其安全性进行了证明。最后，通过实验仿真，与现有低轨卫星身份认证方案相比，所提方案的认证时延、计算开销和通信开销较低。

关键词：低轨卫星网络；身份认证；无证书；随机预言模型
中图分类号：TP309.2　　doi: 10.19734/j.issn.1001-3695.2022.02.0102

Efficient certificateless authentication scheme for leo satellite networks

Zhang Yi, Wu Qi†, Zhou Shuangshuang, Jia Mengzhao
(*School of Communication & Information Engineering*, *Chongqing University of Posts & Telecommunications*, *Chongqing* 400065, *China*)

Abstract: Aiming at the problems of long time delay and high computation cost of using complex bilinear mapping in the current LEO satellite network authentication scheme, this paper proposed an efficient certificateless authentication scheme based on the Gayathri's certificateless authentication scheme. This scheme unifies the user's public key and real identity, so that need not the third party to participate in the authentication process and reduce the authentication delay; uses a small number of point multiplication and point addition operations on the elliptic curve to construct the authentication message, which does not involve the bilinear mapping and reduces the computation cost. Then this paper proves the authentication scheme's security based on the assumption of elliptic curve discrete mathematics under the random oracle model. Finally, through experimental simulation, comparing with the existing LEO satellite identity authentication schemes, the proposed scheme has lower authentication delay, computational overhead and communication overhead.

Key words: LEO satellite network; identity authentication ; certificateless; random oracle model

0　引言

随着社会经济的发展，传统的地面网络已经无法满足人们在海洋、沙漠和深山等特殊区域的通信需求。而卫星网络具有覆盖面积广、通信距离远、不受地理条件限制的特点，可以有效弥补地基网络的不足。其中低轨(Low Earth Orbit, LEO)卫星网络具有低时延、通信功耗小、机动性高等优势[1]，在卫星网络占据着越来越重要的作用。

LEO 卫星网络具有节点暴露、信道开放、资源受限、网络拓扑结构高动态变化、用户终端海量等特点，导致用户接入 LEO 卫星网络容易遭受欺骗、恶意拦截、信息窃取等问题。因此 LEO 卫星网络的安全接入认证问题成为了人们关注的焦点。不同于地面网络，卫星网络节点资源和计算能力受限，无法承担复杂的计算和高额的通信开销。并且，相对于中轨和高轨卫星，用户在使用 LEO 卫星系统各项服务时，对于传输时延、实时性和丢包情况要求更高，所以认证时延不能太高。同时，LEO 卫星链路切换更加频繁，过大的认证时延可能导致用户与原卫星的连接失效后还没与其他卫星建立连接，从而连接中断，影响用户通信质量。所以，认证方案的设计应该在保证安全性的前提下，尽可能满足较小的计算开销、较低的认证时延低和通信成本。

针对 LEO 卫星网络安全身份接入问题。早期 Cruickshank 等人[2]提出了一种基于公钥基础设施(Public Key Infrastructure, PKI)的安全认证协议，其核心思想是通过通信双方的公私钥完成身份认证，但是认证过程中需要复杂的流程交互和计算开销，而且存在证书难以管理的问题。为了避免 PKI 体制的弊端，2015 年 Zhang 等人[3]提出基于异或哈希的轻量认证方案，该方案通过用户到卫星到地面网络控制中(Network Control Center, NCC)的方式完成三方认证，其中卫星在认证过程起到中转作用而不参与计算。虽然认证过程中计算开销很小，但是安全性低，认证过程中消息容易被破解。文献[4,5]提出基于哈希和对称加密的匿名认证方案。方案使用对称加密防止认证消息被破解，同时通过异或哈希对终端设备身份进一步隐藏，提高了安全性，但是仍然存在无法抵御服务器欺骗等攻击。为了进一步提高安全性，文献[6,7]基于椭圆曲线密码体制(Elliptic Curve Cryptography, ECC)设计了一种安全强度高的认证方案，并且通过安全形式化证明该类方案具有抵御各种已知攻击能力。但是，文献[3~7]在认证过程中均需要 NCC 的参与，导致 NCC 负载过大，容易出现单点故障问题。同时认证过程中存在多轮信息交互，导致认证时延过大。随着星上处理能力的不断提高，2020 年赵等人[8]在基于身份密码体制(Identity-Based Cryptography, IBC)上，提出

收稿日期：2022-02-28；**修回日期：**2022-04-23

作者简介：张毅(1970-)，男，重庆万州人，教授，硕士，主要研究方向为卫星网络技术、信息安全；吴奇(1995-)，男(通信作者)，四川达州人，硕士研究生，主要研究方向为卫星网络技术、密码学(s190101171@stu.cqupt.edu.cn)；周霜霜(1997-)，女，重庆江津人，硕士研究生，主要研究方向为无线定位、网络与信息安全；贾梦朝(1996-)，男，河南南阳人，硕士研究生，主要研究方向为物联网、网络与信息安全．

一种用户终端和卫星间端到端的认证方案，该方案仅在注册阶段需要密钥生成中心(Key Generation Center, KGC)生成公私钥，而在认证过程中，不需要第三方参与，从而降低了认证时延。但是其认证方案中使用了复杂的双线性映射，导致计算开销过大，同时由于 KGC 保存着所有用户终端设备的公私钥，所以存在密钥托管问题。

无证书密码体制[9](Certificateless Public Key Cryptography, CL-PKC)可以很好地避免密钥托管问题，同时具有很高的安全性。所以在低轨卫星网络身份认证应用中具有很好的前景。但是现有低轨卫星中无证书认证方案很少，且主要关注在密钥协商[10]和特定卫星系统中[11]。

2017 年，周等人[12]提出一种高效的无证书签密方案，该方案避免了计算量大的双线性运算，而使用少量群上点乘和点加运算，从而降低了计算开销，适用于计算资源有限的设备。但是文献[12]被文献[13]证明了存在安全缺陷，不能抵御伪造身份攻击。近年来，无证书认证方案广泛运用了其他资源受限的网络中，如物联网[14]和车联网[15,16]。对低轨卫星网络中采用无证书认证的方案设计提供了很好的研究思路。

综述，现有认证方案无法满足 LEO 卫星高效认证需求。因此，根据以上研究基础，本文在 Gayathri 等人方案上，结合低轨卫星网络特点，提出了一种高效无证书身份认证方案。该方案中用户认证过程中不需要 NCC 参与，降低了认证时延；同时通过椭圆曲线上少量点乘和点加运算构建认证消息，计算开销低且安全性高。最后在随机预言模型下证明了方案的安全性，并和已有方案性能进行了对比分析。

1　预备知识

1.1　困难问题

椭圆曲线离散数学对问题(Elliptic Curve Discrete logarithm, ECDLP)。令 P 是阶为大素数 q 的循环群 G 的一个生成元；而任意的概率多项式算法 A 成功解决 ECDLP 问题概率 $Adv^{DL}(A)=\Pr[A(P,s_mP=s_m)]$ 是可忽略的，概率来源于 s_m 在 Z_q^* 上随机选取和算法 A 的随机选择。

1.2　无证书认证方案流程

无证书认证方案主要包括以下五个阶段：初始化阶段(Setup)、秘密值生成阶段(Set-secret-key)、部分密钥生成阶段(Set-private-key)、签名阶段(Sign)和验证阶段(Verify)。

初始化(Setup)：该阶段由 KGC 执行，以安全参数 k 作为输入，生成系统的主密钥 s 和系统公共参数 $params$。秘密值生成(Set-secret-key):该阶段由用户终端设备执行，以用户设备 ID 作为输入，生成用户终端秘密值 x 和公开参数 X。部分密钥生成(Set-private-key)：该阶段由 KGC 执行，以用户设备 ID 和公开参数 X 作为输入，生成用户部分密钥 (R,d)。签名(Sign)：该阶段由任意合法设备执行，输入任意消息 m 和该设备的私钥对 (x,d)，生成该消息的签名 σ。验证(Verify)：该阶段由任意合法设备执行，输入任意消息 m 和签名 σ，输出验证结果(true/false)。

1.3　Gayathri 等人方案回顾

令签名者 V 和验证者 R 为无证书认证方案的主要实体。按照 Gayathri 等人方案，则 V 生成的公私钥对为 $<PK_i=(X_i,R_i),SK_i=(d_i,x_i)>$。$PK_i$ 由 PKG 生成部分公钥 R_i 和自己生成的部分公钥 X_i 组成。同理，SK_i 是由 PKG 生成部分私钥 d_i 和自己生成的部分私钥 x_i 组成。

签名阶段(Sign)：V 选取随机数 $y_{1i},y_{2i}\in Z_q^*$，获取当前时间 t_i，然后按照如下方式计算部分密钥 Y_{1i},Y_{2i} 和签名 w_i，然后发送认证请求 $m_i,\sigma_i=(R_i,Y_{1i},u_i,w_i)$ 到验证者 R。

$$Y_{1i}=y_{1i}P,$$
$$Y_{2i}=[(y_{2i}x_i+h_{2i}d_i)\bmod q]P_{pub}=(u_i,v_i),$$
$$w_i=[u_i(y_{1i}+h_{3i}x_i)+h_{4i}d_i]\bmod q$$

其中，$h_{2i}=H_2(m_i,ID_i,Y_{1i})$、$h_{3i}=H_3(m_i,ID_i,Y_{1i},R_i,t_i)$、$h_{4i}=H_4(m_i,ID_i,Y_{1i},R_i,t_i)$ 表示摘要值；(u_i,v_i) 表示椭圆曲线上一点的横纵坐标。P、P_{pub}、q 分别表示椭圆曲线生成元、系统公钥和大素数。

验证阶段(Verify)：R 收到 V 的认证请求后，验证式(1)是否成立，若成立，则表示认证成功，否则，认证失败。

$$w_iP-u_i(Y_{1i}+h_{3i}X_i)=h_{4i}(R_i+h_{1i}P_{pub}) \tag{1}$$

1.4　Gayathri 等人方案安全性缺陷

根据文献[9]描述，无证书密钥体制存在 $\mathcal{A}_1$ 类敌手，该类敌手具有替换合法用户公钥的能力,但它不掌握系统主密钥。当 $\mathcal{A}_1$ 获取 V 公钥 PK_i 后，然后伪造公钥替代 PK_i，生成伪造签名 σ。$\mathcal{A}_1$ 与 R 的具体交互过程如下：

签名(Sign)：$\mathcal{A}_1$ 通过公开信道获取 V 的公钥 $PK_i=(X_i,R_i)$ 和身份标识符 ID_i 后:

获取当前时间 t_i'，选取随机数 $y_{1i}'\in Z_q^*$，计算:

$$Y_{1i}'=y_{1i}'P=(u_i',v_i')$$
$$h_{1i}=H_1(ID_i,R_i,P_{pub})$$
$$h_{3i}'=H_3(m_i',ID_i,Y_{1i}',R_i,t_i')$$
$$h_{4i}'=H_4(m_i',ID_i,Y_{1i}',R_i,t_i')$$

伪造公钥为 $PK_i=(X_i',R_i)$，并替换成 V 的公钥。其中 $X_i'=h_{3i}'^{-1}[-u_i'Y_{1i}'-h_{4i}(R_i+h_1P_{pub})]+Y_{1i}'$。

然后生成签名 $w_i=y_{1i}'$ 最后发送认证请求($m_i',\sigma_i=(R_i,Y_{1i}',u_i',w_i')$)到 R。

验证(Verify)：R 收到 V 的认证请求后：

a)计算：

$$h_{3i}'=H_3(m_i',ID_i,Y_{1i}',R_i,t_i')$$
$$h_{4i}'=H_4(m_i',ID_i,Y_{1i}',R_i,t_i')$$
$$h_{1i}=H_1(ID_i,R_i,P_{pub})$$

b) 验证式(2)是否成立，若成立，则表示认证成功，否则，认证失败。

$$w_i'P-u_i'(Y_{1i}+h_{3i}'X_i')=h_{4i}'(R_i+h_{1i}P_{pub}) \tag{2}$$

因为敌手 $\mathcal{A}_1$ 伪造 V 的签名 σ_i 满足式(1)，所以可以通过 R 的验证。因此 $\mathcal{A}_1$ 具有伪造成合法用户的能力，Gayathri 等人方案无法满足所声称的对 $\mathcal{A}_1$ 敌手的不可伪造性。证明如下：

$$h_{4i}'(R_i+h_{1i}P_{pub})+u_i'(Y_{1i}+h_{3i}'X_i')=$$
$$h_{4i}'(R_i+h_{1i}P_{pub})+u_i'Y_{1i}+h_{3i}'X_i'=Y_{1i}=y_{1i}'P=w_i'P$$

2　本文方案

2.1　系统模型

方案的系统模型如图 1 所示，下面对低轨网络认证中主要实体做简要介绍。

a) KGC：所有实体信息的管理者。它发布公开系统参数，并负责设备信息注册，为合法设备生成公私钥对。

b) LEO 卫星：卫星网络节点，距离高度一般为 500-2000km，传播时延一般在 20~40ms 左右，主要负责用户的接入和数据的传输。本方案中卫星有一定计算能力，具备验证用户合法性的功能。

c) 用户终端：主要包括手机、车辆、飞机、用户等地面终端，需要通过低轨卫星网络获取相应的服务。

2.2　方案流程

本文方案主要包括 4 个阶段：系统初始化阶段、秘密值生成阶段、部分私钥生成阶段、双向认证阶段。符号及其含义如表 1 所示。

2.2.1 系统初始化阶段

该阶段由 KGC 执行，KGC 选取阶数为 q 的循环群 G，其中 q 为大素数($q>2^k$， k 为安全参数)，P 为 G 的一个生成元。定义：$H_1:\{0,1\}^{l_u}\times G\times G\rightarrow Z_q^*$， $H_2:\{0,1\}^*\rightarrow Z_q^*$， l_u 为用户身份标识符 ID 的长度。然后 KCG 随机选取系统主密钥 $s_m\in Z_q^*$，

计算系统公钥 $P_{pub}=s_mP$ 。然后公开参数 $params=<q,P,G,P_{pub},H_1,H_2>$ 。

表 1　符号及其含义

Tab. 1　Symbols and their meanings

符号	含义
q	大素数
Z_q^*	小于 q 的正整数
s_m	系统主密钥
$params$	系统公开参数
P_{pub}	系统公钥
$ID_i,i\in\{U,S\}$	用户和卫星的身份标识符
$PK_i=(X_i,R_i),i\in\{U,S\}$	用户和卫星的的公钥对
$SK_i=(x_i,d_i),i\in\{U,S\}$	用户和卫星的的私钥对
$Q_i,i\in\{U,S\}$	用户和卫星部分密钥
H_1,H_2	单向哈希函数
r,a_1,s_1	Z_q^* 上的随机数
m	消息
$\sigma_i,i\in\{U,S\}$	用户和卫星对消息的签名
$(u_1,v_1),(u_2,v_2)$	椭圆曲线上一点横轴坐标
n,w,α_1,α_2	中间变量

2.2.2 秘密值生成阶段

该阶段由用户终端和卫星分别执行。

设用户终端设备为 ID_U ， ID_U 随机选取秘密值 $x_U\in Z_q^*$，计算公开参数 $X_U=x_UP$ ，然后通过安全通道发送 (ID_U,X_U) 到 KGC。

设卫星设备为 ID_S ， ID_S 随机选取秘密值 $x_S\in Z_q^*$ ，计算公开参数 $X_S=x_SP$ ，然后通过安全通道发送 (ID_S,X_S) 到 KGC。

2.2.3 部分密钥生成阶段

该阶段由 KGC、用户终端和卫星分别执行。

KGC 收到用户终端设备身份标识符 ID_U 和公开参数 X_U ，然后随机选取 $r_U\in Z_q^*$ ，并计算 $R_U=r_UP$ ， $d_U=r_U+s_mH_1(ID_U,X_U,R_U)$ 。然后通过安全信道将 (R_U,d_U) 返回给设备；

用户终端收到 KGC 返回密钥后，验证 $d_UP=R_U+P_{pub}H_1(ID_U,X_U,R_U)$ 是否成立判断 KCG 生成密钥的合法性。若验证不通过，则重新向密钥 KGC 申请密钥。否则，密钥生成成功。最终，用户终端 ID_U 的公钥为 $PK_U=<X_U,R_U>$ ，私钥为 $SK_U=<x_U,d_U>$ 。

同理，卫星 ID_S 的公钥为 $PK_S=<X_S,R_S>$ ，私钥为 $SK_S=<x_S,d_S>$ 。

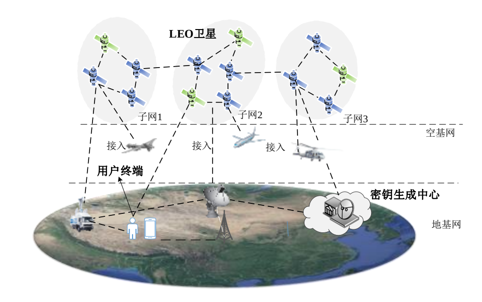


图 1　低轨卫星网络系统模型

Fig. 1　LEO satellite network system model

2.2.4 双向认证阶段

用户 U(身份标识符为 ID_U)和卫星 S(身份标识符 ID_S)进行双向身份认证流程如下：

用户 U 随机选取 $a_1\in Z_q^*$

计算 $Q_U=a_1P=(u_1,v_1)$ ，时间戳 T_U ；

计算 $n_U=H_2(ID_U,X_U,T_U,R_U,Q_U,m)$ ；

然后生成签名 $\sigma_U=a_1^{-1}(n_Ux_U+u_1d_U)\bmod q$ ；

发送认证请求 $req=(ID_U,X_U,R_U,Q_U,T_U,\sigma_U,m)$ 。

卫星 S 收到用户发送的认证请求 req 后，获取当前时间 T ，判断 $T-T_U\le\Delta T$ 是否成立，若不成立则认为消息不新鲜，然后丢弃消息并停止操作。否则：

计算 $w=\sigma_U{}^{-1}$ ， $h_U=H_1(ID_U,X_U,R_U)$ ， $n_U'=H_2(ID_U,X_U,T_U,R_U,Q_U,m)$ ；

计算 $\alpha_1=n_U'w,\alpha_2=u_1w$ ；

验证 $Q_U=\alpha_1X_U+\alpha_2(R_U+h_UP_{pub})$ 。

若卫星 S 验证用户 U 身份合法性通过后，卫星 S 随机选取 $s_1\in Z_q^*$

计算 $Q_S=s_1P=(u_2,v_2)$ ，时间戳 T_S ；

计算 $n_S=H_2(ID_S,X_S,T_S,R_S,Q_S)$ ；

然后生成签名 $\sigma_S=a_1^{-1}(n_Sx_S+u_2d_S)\bmod q$ ；

返回认证响应 $res=(ID_S,X_S,R_S,Q_S,T_S,\sigma_S)$ 。

用户收到卫星返回的认证响应后，获取当前时间 T ，判断 $T-T_U\le\Delta T$ 是否成立，若不成立则认为消息不新鲜，认证失败。否则：

计算 $h_S=H_1(ID_S,X_S,R_S)$ ， $w=\sigma_S{}^{-1}$ ， $n_S'=(ID_S,X_S,T_S,R_S,Q_S)$ ；

计算 $\alpha_1=n_S'w,\alpha_2=u_2w$ ；

验证 $Q_S=\alpha_1X+\alpha_2(R_S+h_SP_{pub})$ 。

如果用户 U 和卫星 S 顺利进行完成以上步骤，则认为用户和卫星双向认证成功。

3　安全性证明与分析

3.1　正确性证明

卫星在认证请求阶段判断签名是否成立的正确性分析如下：

如果用户签名合法，那么签名需要满足等式 $Q_U=\alpha_1X_U+\alpha_2(R_U+h_UP_{pub})$ ；

由用户秘密值生成阶段和部分密钥生成阶段可知：用户密钥满足等式 $d_UP=R_U+P_{pub}H_1(ID_U,X_U,R_U)$ 、 $X_U=x_UP$ 。所以卫星可以通过计算得到的 h_U 和公开参数 X_U 、 R_U 、 P_{pub} 计算出 x_UP 和 d_UP ；

a) 进一步，卫星可进行如下计算：

$$\alpha_1X_U+\alpha_2(R_U+h_UP_{pub})=\alpha_1x_UP+\alpha_2d_UP=$$

$$(\alpha_1x_U+\alpha_2d_U)P=w(n_U'x_U+u_1d_U)P=\sigma_U{}^{-1}(n_U'x_U+u_1d_U)P$$

而用户在生成签名时，签名 σ_U 满足等式 $\sigma_U=a_1^{-1}(n_Ux_U+u_1d_U)$ ，带入上一步，就可以得到 $\alpha_1X_U+\alpha_2(R_U+h_UP_{pub})=a_1P=Q_U$ 。

3.2　安全性正式证明

定理 1　假定敌手 $\mathcal{A}_1$ 在多项式时间内，以无法忽视的优势 ζ_1 成功伪造出一个签名，则挑战者 $\mathcal{C}$ 能够以 $(1-\frac{q_p}{q})(1-\frac{q_s}{q})(\frac{q_c}{q_c+1})^{q_c}\zeta_1$ 解决 ECDLP 问题。其中 q_p 表示部分密钥生成查询次数， q_s 表示秘密值查询次数， q_c 表示次签名查询次数。

证明　假设一个敌手 $\mathcal{A}_1$ 能在本方案中能以优势 ζ_1 成功伪造目标用户 ID_i 的有效签名。则对于给定的 $(P,Q=aP)$ ，挑战者 $\mathcal{C}$ 的目标是计算出 a 。

挑战者 $\mathcal{C}$ 与 $\mathcal{A}_1$ 的交互流程如下：

初始化阶段： $\mathcal{C}$ 首先运行 Setup 程序构建系统，令 $P_{pub}=s_mP$ ，然后公开系统的参数 $params=<q,P,G,P_{pub},H_1,H_2>$ 。建立并维护 L_1,L_2,L_P,L_s,L_k 5 个表，哈希表 L_1 填充内容为 (ID,X,R,h_1) ，哈希表 L_2 填充内容为 (ID,X,R,T,Q,m,h_2) ，部分密钥值表 L_P 填充内容为 (ID,R,d) ，秘密值表 L_s 填充内容为 (ID,x) ，公钥表 L_k 填充内容为 (ID,X,R) 。初始化时，各个表格内容为空。

预言机查询阶段：本阶段 $\mathcal{A}_1$ 与 $\mathcal{C}$ 之间进行如下预言机交互：

L_1 查询：当 $\mathcal{A}_1$ 输入 (ID,X_i,R_i) 时进行询问时。若 L_1 存在相应的元组 (ID,X_i,R_i,h_1) ，则返回 h_1 给 $\mathcal{A}_1$ ；否则，则 $\mathcal{C}$ 随机选取 $h_1\in Z_q^*$ ，然后返回 h_1 给 $\mathcal{A}_1$ ，并保存 (ID,X_i,R_i,h_1) 。

L_2 查询：当 $\mathcal{A}_1$ 输入 (ID,X_i,R_i,T_i,Q_i,m_i) 进行询问时。若 L_2 存在相应的元组 $(ID,X_i,R_i,T_i,Q_i,m_i,h_2)$ ，则返回 h_2 给 $\mathcal{A}_1$ ；否则，则

$\mathcal{C}$ 随机选取 $h_2 \in Z_q^*$，然后返回 h_2 给 $\mathcal{A}_1$，并保存 $(ID, X_i, R_i, T_i, Q_i, m_i, h_2)$ 到 L_2 中。

部分密钥生成查询：当 $\mathcal{A}_1$ 输入 (ID, X_i) 进行询问时，首先判断 $ID = ID*$ 是否成立，若成立，则 $\mathcal{C}$ 终止模拟。若不成立，则判断 L_P 是否存在相应的元组 (ID, R_i, d_i)，若存在，则返回 (R_i, d_i) 给 $\mathcal{A}_1$，否则 $\mathcal{C}$ 随机选取 $d_i, h_1 \in Z_q^*$，并计算 $R_i = d_i P - h_1 P_{pub}$，然后返回 (R_i, d_i) 给 $\mathcal{A}_1$。最后，保存 (ID, R_i, d_i) 到 L_p 中，并添加 (ID, X_i, R_i, h_1) 到 L_1 中。

秘密值查询：当 $\mathcal{A}_1$ 输入 ID 进行询问时，首先判断 $ID = ID*$ 是否成立，若成立，则 $\mathcal{C}$ 终止模拟。若不成立，则判断 L_s 是否存在相应的元组 (ID, x_i)，若存在，则返回 x_i 给 $\mathcal{A}_1$，否则 $\mathcal{C}$ 随机选取 $x_i \in Z_q^*$，然后返回 x_i 给 $\mathcal{A}_1$ 然后保存 (ID, x_i) 到 L_s 中。

公钥生成查询：当 $\mathcal{A}_1$ 输入 (ID, m_i) 进行询问时，若 L_k 存在相应的元组 (ID, X_i, R_i)，则返回 (X_i, R_i) 给 $\mathcal{A}_1$，否则，$\mathcal{C}$ 随机选取 $x_i \in Z_q^*$，计算 $X_i = x_i P$，通过 (ID, X_i) 对 L_P 进行查询得到 (R_i, d_i)，然后返回 (ID, X_i, R_i) 给 $\mathcal{A}_1$，并分别添加 (ID, R_i, d_i)，(ID, X_i) 到 L_P 和 L_s 中。

公钥替换：$\mathcal{A}_1$ 可以选择一个新的公钥 (X_i', R_i) 替换任意合法用户公钥 (X_i, R_i)，并保存到 L_P 中。

签名查询：当 $\mathcal{A}_1$ 以 (ID, m_i) 进行询问时，首先判断 $ID = ID*$ 是否成立，若成立，则 $\mathcal{C}$ 终止模拟。不然，随机选取 $a_i \in Z_q^*$，计算 $Q_i = a_i P = (u_i, v_i)$，然后分别通过 L_2 L_P L_s 分别获取 h_2、d_i、x_i，计算 $\sigma_i = a_i^{-1}(h_2 x_i + u_i d_i) \bmod q$，返回 (Q_i, σ_i) 给 $\mathcal{A}_1$。

初始化阶段：本阶段 $\mathcal{A}_1$ 通过 $\mathcal{C}$ 之间的交互后，尝试伪造合法用户签名。

伪造：最后 $\mathcal{A}_1$ 伪造出 $ID*$ 的一个签名 (Q_i^*, σ_i^*)，如果等式(3)成立，则 $\mathcal{A}_1$ 伪造成功。其中 u_i^* 为 Q_i^* 的一部分。

$$h_2^* \sigma_i^{*-1} X_i^* + u_i^* \sigma_i^{*-1}(R_i^* + h_1^* P_{pub}) = Q_i^* \tag{3}$$

根据分叉引理[15]，$\mathcal{A}_1$ 能够在多项式时间内以同样的方式选择不同的 h_1' 成功伪造另一个签名 (Q_i^*, σ_i') 满足等式(4)。

$$h_2^* \sigma_i'^{-1} X_i^* + u_i^* \sigma_i'^{-1}(R_i^* + h_1' P_{pub}) = Q_i^* \tag{4}$$

所以由等式(3)(4)可得

$$Q_i^*(\sigma_i^{*-1} - \sigma_i'^{-1}) = u_i^*(h_1^* - h_1') P_{pub}$$
$$a_i^*(\sigma_i^{*-1} - \sigma_i'^{-1}) P = u_i^*(h_1^* - h_1') s_m P$$

$\mathcal{C}$ 可计算出 $s_m = a_i^*(\sigma_i^{*-1} - \sigma_i'^{-1}) u_i^{*-1}(h_1^* - h_1')^{-1}$。

概率分析：设在多项式时间内，$\mathcal{C}$ 最多进行 q_p 次部分密钥生成查询，q_s 次秘密值查询，q_c 次签名(即选取了 q_c 个身份进行签名询问，1 个身份作为挑战即 $ID*$)后，成功解决 ECDLP 问题，那么 $\mathcal{C}$ 在模拟过程中必须满足以下三个条件：

E_1：签名阶段未终止(未选中 $ID*$)。

E_2：$\mathcal{C}$ 未对目标身份号($ID*$)进行部分密钥生成查询或秘密值查询。

E_3：σ 是关于 $ID*$ 的一个有效签名。

所以，优势可以表示为 $P_r[E_1 \wedge E_2 \wedge E_3]$。而 $P_r[E_1]=(1-\frac{1}{q_c+1})^{q_c}$、$P_r[E_1 | E_2]=(1-\frac{q_p}{q})(1-\frac{q_s}{q})$、$P_r[E_3 | E_1 \wedge E_2]=\zeta_1$。所以模拟过程不终止的概率为 $P_r[E_1 \wedge E_2 \wedge E_3] = (1-\frac{q_p}{q})(1-\frac{q_s}{q})(\frac{q_c}{q_c+1})^{q_c} \zeta_1$。

定理 2　假定敌手 $\mathcal{A}_2$ 在多项式时间内，以无法忽视的优势 ζ_2 能够成功伪造出一个签名，则挑战者 $\mathcal{C}$ 能够以 $(1-\frac{q_p}{q})(1-\frac{q_s}{q})(\frac{q_n}{q_n+1})^{q_n} \zeta_2$ 解决 ECDLP 问题，其中 q_p 表示公钥生成查询次数，q_s 表示秘密值查询次数，q_n 表示次签名查询次数。

证明　证明过程和定理 1 类似。

3.3　安全性形式化分析

签名不可链接性：在本文方案中，用户终端和卫星每次生成签名 $\sigma_i = a_i^{-1}(n_i x_i + u d_i) \bmod q$ 都需要随机选取的 a_i，导致每次生成签名都是随机的，因此签名之间没有任何联系，具有签名不可链接性。

前向安全性：前向安全性可以保障用户终端前后的认证消息不会互相影响。本文方案中，当用户终端 $SK_U =< x_U, d_U >$ 遭到破坏时，也不会泄露先前建立的会话密钥信息。因为会话密钥 $k = a_i Q_s = s_i Q_u$ 是基于迪菲-赫尔曼密钥交换(Diffie–Hellman Key Exchange, DHKE)协议生成的，不依赖于用户终端的私钥，因此本文认证方案具有前向安全性。

a) 抗重放攻击：认证过程中，卫星和用户终端都会根据当前的时间戳判断收到消息的新鲜性，若大于系统设定最大忍受时延，则会丢弃消息，从而使攻击者对消息的重放攻击无效。

b) 抵御中间人或假冒攻击：攻击者无法根据截获用户发送的认证消息伪造用户终端的签名。因为认证消息构建的安全基础是基于椭圆曲线上的离散对数问题的困难性，所以攻击者无法伪造认证消息。

密钥托管弹性：在所提出的方案中，用户终端和卫星的私钥 $SK_i =< x_i, d_i >$ 包括由 KGC 计算的部分私钥 d_i 以及用户终端和卫星随机选择的 x_i，因此，恶意的 KGC 在不知道 x_i 的情况下无法生成有效的签名。所以，所提出的方案不受密钥托管问题的影响。

4　实验仿真

计算开销、认证时延和通信开销是衡量身份认证方案优劣的最直观的指标。本节将从计算开销、认证时延和通信开销三方面与近年来发表的、安全性较高的接入认证方案相对比。选取的对比方案分别为：文献[7]提出的基于三因素椭圆曲线认证方案、文献[8]提出的基于双线性配对的认证方案和文献[11]提出的基于双线性配对的无证书认证方案。

4.1　实验环境

本文实验环境操作系统为 Ubuntu 18.04.6 64bit，在 VMware 虚拟机运行，运行内存为 4GB。硬件环境为 Intel i5-10210U 1.60GHz。本文所提的认证方案在 Charm-crypto 库上实现，操作运算基于底层 PCB 库。随机模拟 100 次不同密码运行操作进行分析，得到的结果见表 2。

表 2　不同操作运行时间(ms)

Tab. 2　Running time of different operations(ms)

符号	操作	时间
T_{par}	双线性对运算	5.9297
E_M	椭圆曲线上点乘运算	1.0494
E_A	椭圆曲线上点加运算	0.0016

4.2　计算开销和认证时延分析

在方案对比时，主要考虑 E_M、E_A 和 T_{par} 产生的计算开销，而普通哈希运算和四则运算花费时间很少，因此将其可以忽略。同时，为了不失一般性，设用户终端设备和卫星单向传输时延 T_{u-s} 和卫星和 NCC 单向传输时延 T_{s-n} 都为 20ms。如表 3 所示，本文认证方案不涉到复杂度较大 T_{par} 运算，在签名和验证阶段仅需要少量的 E_M 和 E_A 运算就可以完成验证，计算开销时间大约为 4.2ms。与使用双线性对运算认证方案[8]约 24.1ms 和方案[11]约 10ms 相比，计算开销上有明显优势；与不使用双线性对运算方案[7]相比，方案[7]认证过程中需要较多的 E_M 运算约 6.2ms，计算开销较本文更大，同时相对本文方案相对多了 $2T_{s-n}$ 次交互次数。所以，本文认证方案计算开销最低，认证更加高效。

认证时延主要包括身份认证中的计算开销和认证信息交互过程中传播时延开销。由图 2 不同消息个数下的认证时延可知，由于本文方案使用少量复杂度较低的椭圆曲线上点乘

和点加运算构建签名，计算开销最小；同时认证方案是基于无证书认证模型，不需要第三方参与认证，减少了认证过程中的交互次数，传播时延低。所以本文的认证时延均明显小于其他方案。

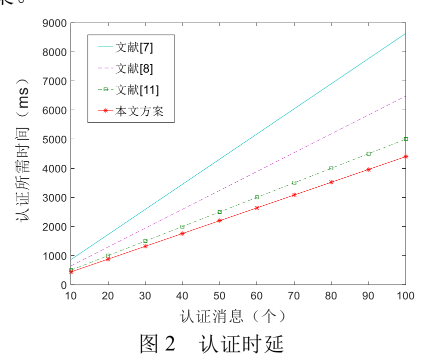


图 2　认证时延

Fig. 2　Authentication delay

表 3　认证开销比较

Tab. 3　Comparison of authentication costs

方法	签名	验证	计算总开销	交互次数
文献[7]	$3\ E_M$	$3\ E_M$	$6\ E_M$	$2\ T_{u-s} +2\ T_{s-n}$
文献[8]	$T_{par} +4\ E_M + E_A$	$3\ T_{par} + E_A$	$4\ T_{par} + E_M +2\ E_A$	$2\ T_{u-s}$
文献[11]	$2\ E_M$	$T_{par} +2\ E_M +2\ E_A$	$T_{par} +4\ E_M +2\ E_A$	$2\ T_{u-s}$
本文方案	E_M	$3\ E_M +2\ E_A$	$4\ E_M +2\ E_A$	$2\ T_{u-s}$

4.3　通信开销分析

为了方便比较，本文假定设备身份标识长度 $|ID|$ 为 4 Byte、时间戳长度 $|T|$ 为 4 Byte，消息长度 $|m|$ 为 20 Byte、正整数域 Z_q^* 中元素占用 $|Z_q^*|$ 为 20 Byte。而循环群 G 和 G_1 占用字节数分别为 20 Byte 和 64 Byte，所以 G 和 G_1 上元素长度 $|G|$ 和 $|G_1|$ 分别为 40 Byte 和 128 Byte。不同认证方案通信成本比较如表 4 所示，可以看出，相比于其他方案，本文认证方案通信成本更低。

5　结束语

本文提出一种面对低轨卫星的高效无证书身份认证方案。该方案中用户认证过程中不需要第三方参与，降低了认证时延；认证过程无须双线性映射，计算开销低，同时具有签名不可链接性、前向安全性、抗重放攻击、抵御中间人或假冒攻击、密钥托管弹性等安全性。但是方案在计算开销上还有一定优化空间，需要用户终端和卫星具备一定计算能力。如何在满足高安全强度的同时，构造计算复杂度更低的低轨卫星网络身份认证方案是未来的重要研究方向。

表 4　通信成本比较

Tab. 4　Comparison of communication costs

方法	通信长度/Byte
文献[7]	$2\|G\|+4\|Z_q^*\|+\|T\|+\|ID\|+\|m\|=188$
文献[8]	$2\|G_1\|+\|T\|+\|ID\|+\|m\|=284$
文献[11]	$3\|G_1\|+\|T\|+\|ID\|+\|m\|=412$
本文方案	$3\|G\|+\|Z_q^*\|+\|T\|+\|ID\|+\|m\|=168$

参考文献：

[1] 杨昕，孙智立，刘华峰，等. 新一代低轨卫星网络和地面无线自组织网络融合技术的探讨 [J]. 中兴通信技术，2016, 22 (4): 58-63. (Yang Xin, Sun Zhili, Liu Huafeng, *et al.* Technology of new generation LEO satellite network and terrestrial MANET integration [J]. ZTE TECHNOLOGY JOURNAL, 2016, 22 (4): 58-63.)

[2] Cruickshank H S. A security system for satellite networks [C]// Fifth International Conference on Satellite Systems for Mobile Communications and Navigation, London: IET Press, 1996: 187-190.

[3] Zhang Yuanyuan, Chen Jianhua, Huang Baojun. An improved authentication scheme for mobile satellite communication systems [J]. International Journal of Satellite Communications and Networking, 2015, 33 (2): 135-146.

[4] Saroj T, Gaba G S. A Lightweight Authentication Protocol based on ECC for Satellite Communication [J]. Pertanika Journal of Science & Technology, 2017, 25 (4): 1317-1330.

[5] 朱辉，陈思宇，李凤华，等. 面向低轨卫星网络的用户随遇接入认证协议 [J]. 清华大学学报：自然科学版，2019, 59 (1): 1-8. (Zhu Hui, Chen Siyu, LI Fenghua, *et al.* User random access authentication protocol for low earth orbit satellite networks. Journal of Tsinghua University (Science and Technology) , 2019, 59 (1): 1-8.)

[6] Qi Mingping, Chen Jianhua, Chen Yitao. A secure authentication with key agreement scheme using ECC for satellite communication systems [J]. International Journal of Satellite Communications and Networking, 2019, 37 (3): 234-244.

[7] Ostad-Sharif A, Abbasinezhad-Mood D, Nikooghadam M. Efficient utilization of elliptic curve cryptography in design of a three-factor authentication protocol for satellite communications [J]. Computer Communications, 2019, 147: 85-97.

[8] 赵国锋，周文涛，徐川，等. 一种基于双线性配对的天地一体化网络安全身份认证方案 [J]. 信息网络安全，2020, 20 (12): 33-39. (Zhao Guofeng, Zhou Wentao, Xu chuan *et al.* A Secure identity authentication scheme for space-ground integrated network based on bilinear pairing [J]. Netinfo Security, 2020, 20 (12): 33-39.)

[9] Al-Riyami S S, Paterson K G. Certificateless public key cryptography [C]// International conference on the theory and application of cryptology and information security. Berlin: Springer Press, 2003: 452-473.

[10] Liu Yuchen, Zhang Aixin, Li Jianhua, *et al.* An anonymous distributed key management system based on CL-PKC for space information network [C]// 2016 IEEE international conference on communications (ICC) . Kuala Lumpur: IEEE Press, 2016: 1-7.

[11] Wu Zhijun, Yang Yiming. BD-D1Sec: Protocol of security authentication for BeiDou D1 civil navigation message based on certificateless signature [J]. Computers & Security, 2021, 105: 102251.

[12] 周彦伟，杨波，王青龙. 安全的无双线性映射的无证书签密机制 [J]. Journal of Software, 2017, 10: 2757-2768. (Zhou Yanwei, Yang Bo, Wang Qinglong. Secure certificateless signcryption scheme without bilinear pairing [J]. Journal of Software, 2017, 28 (10): 2757-2768.)

[13] Yang Xiaodong, Liu Rui, Chen Guilan, *et al.* Security analysis of a certificateless signcryption mechanism without bilinear mapping [C]// 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC) . IEEE, 2020, 1: 2431-2434.

[14] Tedeschi P, Sciancalepore S, Eliyan A, *et al.* LiKe: Lightweight certificateless key agreement for secure IoT communications [J]. IEEE Internet of Things Journal, 2019, 7 (1): 621-638.

[15] Pointcheval D, Stern J. Security arguments for digital signatures and blind signatures [J]. Journal of cryptology, 2000, 13 (3): 361-396.

[16] Gayathri N B, Thumbur G, Reddy P V, *et al.* Efficient pairing-free certificateless authentication scheme with batch verification for vehicular ad-hoc networks [J]. IEEE Access, 2018: 31808-31819.

[17] Xu Guangquan, Zhou Wenjuan A Security-Enhanced Certificateless Aggregate Signature Authentication Protocol for InVANETs, in IEEE Network, IEEE Network [J]. 2020, 34 (2): 22-29.